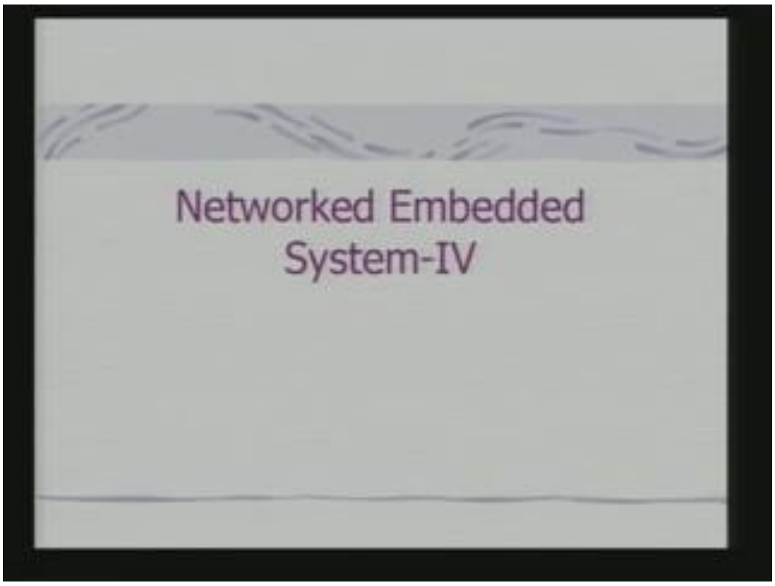
Embedded Systems
Prof. Dr. Santanu Chaudhury
Department of Electrical Engineering
Indian Institute of Technology, Delhi

Lecture – 27
Networked Embedded Systems – IV

So, we were discussing networked embedded systems. In the last class we had discussed how we can combine sensors with embedded system, distribute them all around and link them up using the network. So, today we shall continue on the same discussion.

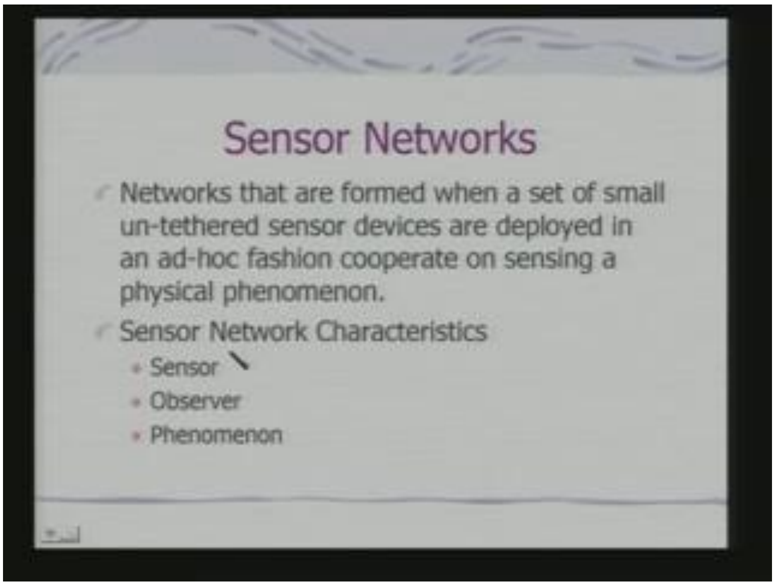
(Refer Slide Time: 01:21)



Networked Embedded System-IV

And also look at the ways means to design network centric embedded system.

(Refer Slide Time: 01:33)

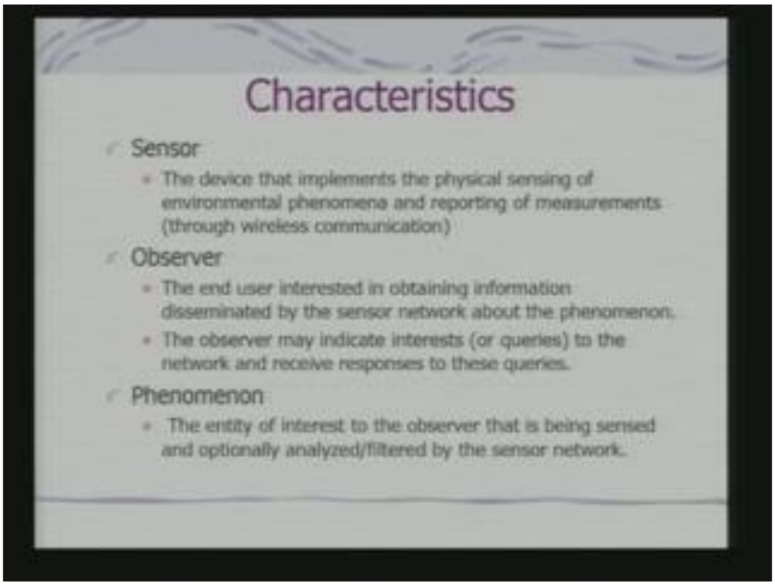


Sensor Networks

- Networks that are formed when a set of small un-tethered sensor devices are deployed in an ad-hoc fashion cooperate on sensing a physical phenomenon.
- Sensor Network Characteristics
  - Sensor
  - Observer
  - Phenomenon

So, just we capitulate sensor networks and networks that are formed when a set of small until that sensor devices are deployed in an ad hoc fashion. And they cooperate on sensing a physical phenomena that physical phenomena may be a natural phenomena may be a man made phenomena. And obviously, a sensor network characteristics a sensor observer and the phenomena which is being monitored.

(Refer Slide Time: 02:01)



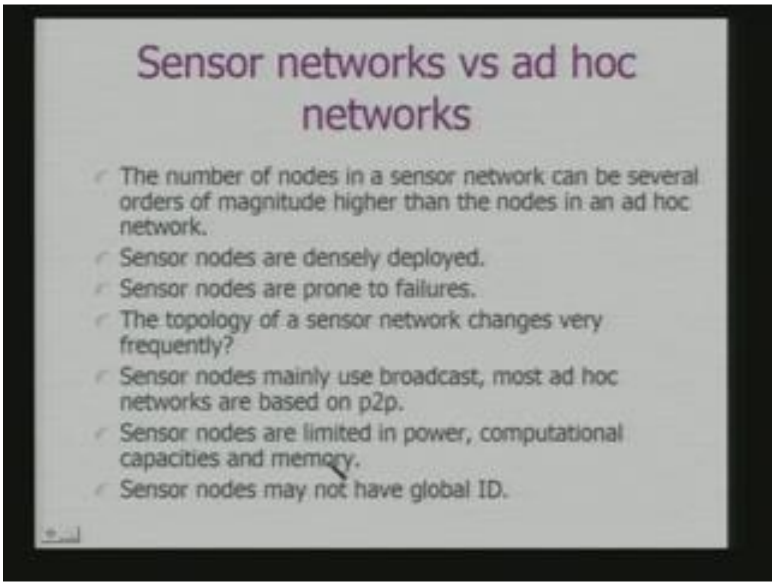
Characteristics

- Sensor
  - The device that implements the physical sensing of environmental phenomena and reporting of measurements (through wireless communication)
- Observer
  - The end user interested in obtaining information disseminated by the sensor network about the phenomenon.
  - The observer may indicate interests (or queries) to the network and receive responses to these queries.
- Phenomenon
  - The entity of interest to the observer that is being sensed and optionally analyzed/filtered by the sensor network.

Sensor is a device that implements a physical sensing of environmental phenomena and reporting of measurements. So, in this case we are generalizing the basic definition of the sensor I hope you realize by now, because a sensor would be actually the actual sensing elements interfaced with an embedded system which as also got the communication element built in to it. We had already seen example of Berkeley mode and Intel node and in the Intel node you have seen that the Bluetooth communication protocol has formed an integral part. But it is not necessary that always this kind of ad hoc networking protocols will be there as part of the mode there can be an additional module plugged in to the sensor node. Observer is a end user was interested in obtaining information disseminated by the sensor network about the phenomena.

Observer may indicate interest or queries to the network and receive responses to this queries and this query can be remotely submitted if you remember the problem of tracking and object there may be multiple objects entering the system or entering the area they monitored. But the query can be generated at the remote node and the basic problem therefore, based on to that of communicating this information back to the query node. So, observer becomes a very important component in the sensor network, because your data communication is requirement is built around the observer's requirement and that application that observer is intending to run on the sensor network. The physical phenomena is the entity of interest to the observer that is being sensed and optionally aliased or filtered for the sensor network, because this analysis and filtering task can be node centrate and need not be centralized.

(Refer Slide Time: 04:20)



Sensor networks vs ad hoc networks

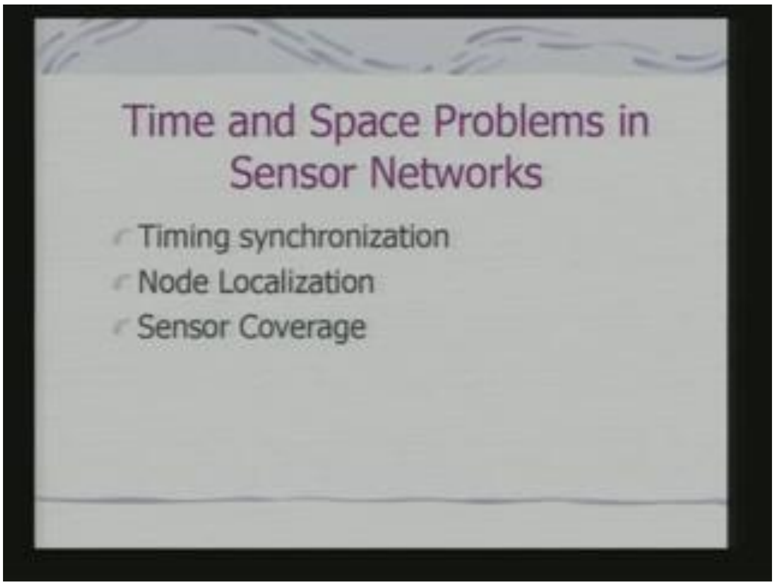
- The number of nodes in a sensor network can be several orders of magnitude higher than the nodes in an ad hoc network.
- Sensor nodes are densely deployed.
- Sensor nodes are prone to failures.
- The topology of a sensor network changes very frequently?
- Sensor nodes mainly use broadcast, most ad hoc networks are based on p2p.
- Sensor nodes are limited in power, computational capacities and memory.
- Sensor nodes may not have global ID.

So, in this background what is the basic difference therefore, from sensor networks and ad hoc networks we have seen a simple example of using an ad hoc protocol like a Bluetooth protocol when used in the sensor network. We can use Bluetooth protocol even we can use 822.11 b in an ad hoc mode with the sensor networks. But by nature sensors networks are different from classical ad hoc network the differences are the following first is the number of nodes in a sensor network can be several orders of magnitude higher than the nodes in the ad hoc network. And sensor nodes are typically densely deployed and ad hoc network can be formed, because of a set of mobile users moving in into a neighborhood at a particular location. But in case of sensor network the sensor nodes are expected to be densely deployed all around the sensor nodes prime to failure. So, there should be mechanisms such that these failures can be notified are taking care of topology of a sensor network does it change very frequently.

It can change depending on how the network is being set up it can change if the nodes themselves have mobility. That means, nodes if they can move around if you consider a set of swam of robots that is a set of robots communicating with each other and moving around then; obviously, the topology is dynamically changing. Sensor nodes mainly used broadcast this is the very basic difference sensor nodes mainly used broadcast most ad hoc networks a based on point to point most ad hoc networks are based on point to point communication while the sensor nodes mainly used broadcast. So, 2 sensor nodes can communicate with each other when they are within a desirable range because if a node is a broadcasting if the other node is within a particular range then only it can listen to the broadcast.

Sensor node are typically limited in power computational capacities and memory, because they are small elements. And they are typically battery power, because if they are to be deployed environment they cannot be provided with electrical wired power. So, for them the energy constraint is much more stricter than that of your PDA or a laptop using which you typically build up. And ad hoc network sensor nodes may not also have a global id just like say computer if it is connected on the internet connected through ip will typically have a IP address. Now, a sensor node may not have an ip address. So, it may not have a global id. So, this are the basic differences of sensor networks from that of a typical ad hoc network and that is why the sensor network problem is different. And requires develop end of different kind of algorithms than that of the classical ad hoc network thing schemes although it is not the ad hoc network is protocol and not being used in the context of sensor networks.

(Refer Slide Time 08:16)



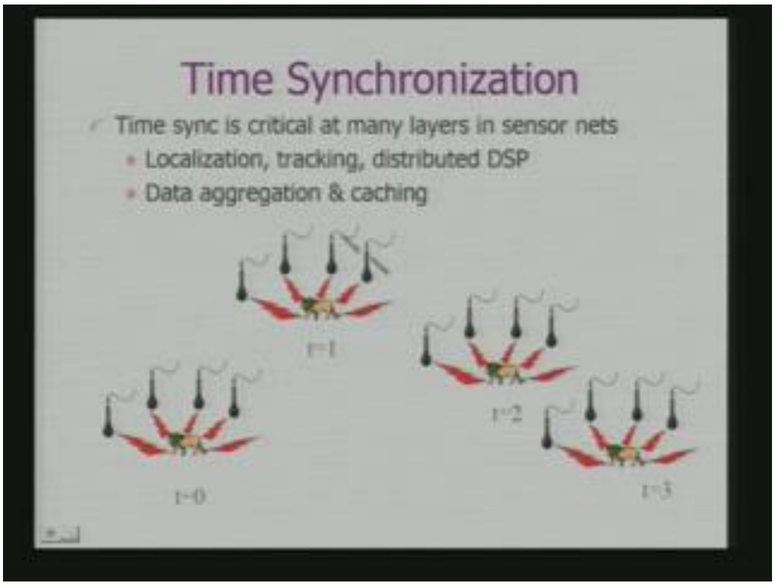
Time and Space Problems in Sensor Networks

- Timing synchronization
- Node Localization
- Sensor Coverage

A very important problem in the context of sensor network is what is called time and space problems. Because you have to have a timing synchronization if an event is detected at a node and if that node has to ask the necks node to observe are track the phenomena then they are has to be a timing synchronization. Because the next node cannot start tracking the phenomena after and you delay when the phenomena has disappear itself if you remember the problem of tracking. So, node A has detected a target when the target moves away from the observations fear of node A node B should take up which node will really take up then is being predicted by node A. And if the prediction is correct the node B should now start observing the phenomena and node B should start observing the phenomena satisfying a timing constraint. The time synchronization becomes important issues; obviously, the other problem is that of node3 localization is node A has detected a target moving out of its fear of influence it should realize in which direction it is moving out.

It would localize the direction in which the target is moving out and along with it we should localize itself in the sensor net in the context of its neighbors. Because it has to identify an appropriate neighbor and inform the neighbor to track the phenomena. If it cannot identify the neighbor if there is an certainty in determination of its localization parameters. Then a certainty should be taken care of should be minimized by instruction by more than 1 node to track the phenomena. These aspects become critical important for a sensor network associated with this sensor coverage, because if I have an area which is to be covered. Then I need to know that what is really my coverage area because then only I can sends whether my task is over or not and what can I do as long as the phenomena is with in my area or coverage sensing coverage. So, all this issues this are not relevant in the context of an pure ad hoc network. Because in a pure ad hoc network the problem is that of simple setting up in a communication link in ad hoc fashion here along with the communication the fundamental problem is that of sensing and processing of the sensed data. So, these makes the sensor network protocols different.

(Refer Slide Time: 11:28)



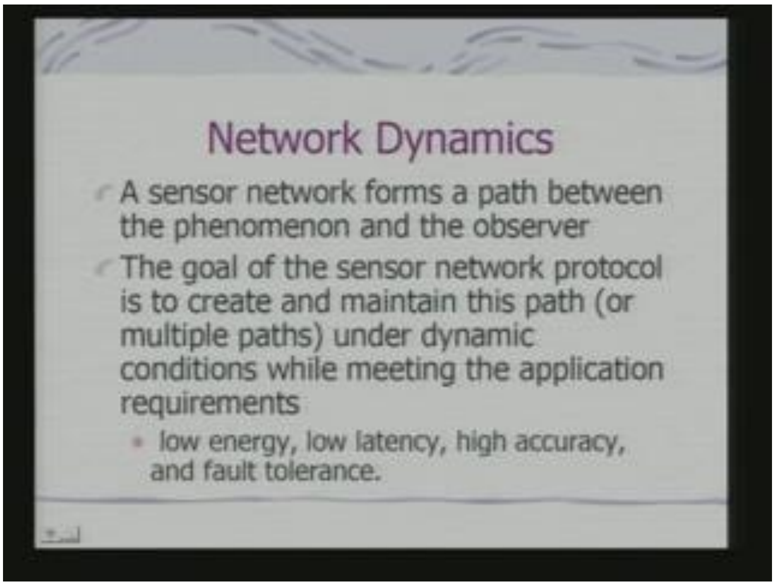
Time Synchronization

- Time sync is critical at many layers in sensor nets
  - Localization, tracking, distributed DSP
  - Data aggregation & caching

So, let us look at this example if I am considered ring or bothering about the problem of tracking element. So, actually if you look in to it is animal are the target is a mobile target it is moving around. And there are this many sensors which may are trying to pick up this information now as the tiger is moving are this is not the tiger this as lion is moving around. So, the whole question comes up in is that of what is that trajectory which is being followed. So, if you see here it is t 0 t 1 t 2 t 3 fine then there has to a time synchronization between the nodes to really track the trajectory of the target. So, what we need we need a localization tracking and distributed DSP why distributed DSP, because I need to do a signal processing to identify this is really the target that I would like to observe.

And this is related to data aggregation because the track can immerge aggregate the data over distributed sensors that is why data aggregation, becomes an important problem and I can use cashing in a very interesting way. So, if I know that this is the target that the fact the target may have certain characteristics. So, that characteristics that I can pass on to the next node then detection of the target can become easier the time delay for the target detection reduces. So, that is effectively becomes a problem of cashing the target characteristics from node to node for the purpose of tracking. So, this; obviously, the whole thing and work provided we can have time synchronization as well as node localization. So, what is the networks dynamics? Therefore, this network is basically has got its own dynamics which is different from any other communication.

(Refer Slide Time: 13:44)

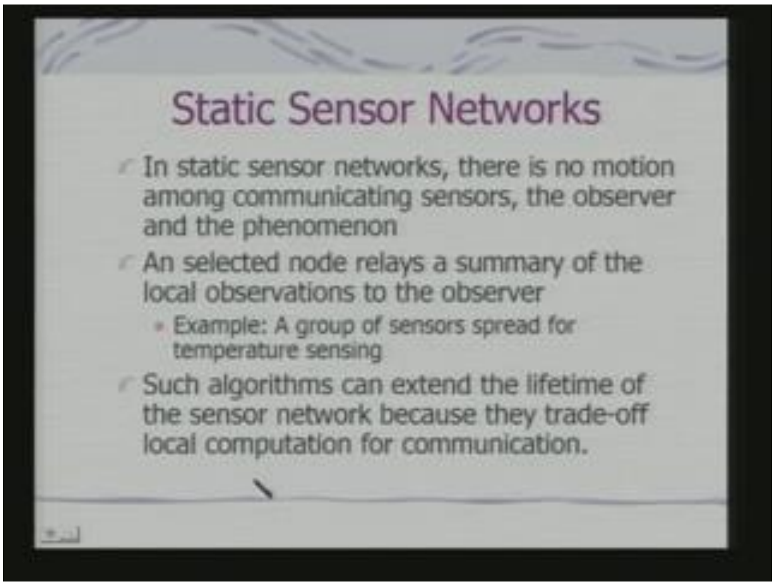


Network Dynamics

- A sensor network forms a path between the phenomenon and the observer
- The goal of the sensor network protocol is to create and maintain this path (or multiple paths) under dynamic conditions while meeting the application requirements
  - low energy, low latency, high accuracy, and fault tolerance.

Network dynamics; very simply stated a sensor network forms a path between the phenomena and the observer why, because observer in this case if you find out the server is located somewhere and observer is located somewhere and observer is interested to know the track of the target. So, sensor network is basically providing the path the observer may fare query at a particular node in that case actually the path gets determined that is the I may track the target using nodes abc and the query may be fared at node G. So, the data aggregated should be transmitted to node G. So, therefore, the goal of sensor network protocol is to create and maintain this path are multiple paths under dynamic conditions while meeting the application requirements. And this requirements; obviously, decide generic requirements low energy, low latency, high accuracy and fault tolerance. Because there may be a failures and I would need to get the track the spied the failure of 1 or 2 nodes.

(Refer Slide Time: 14:53)



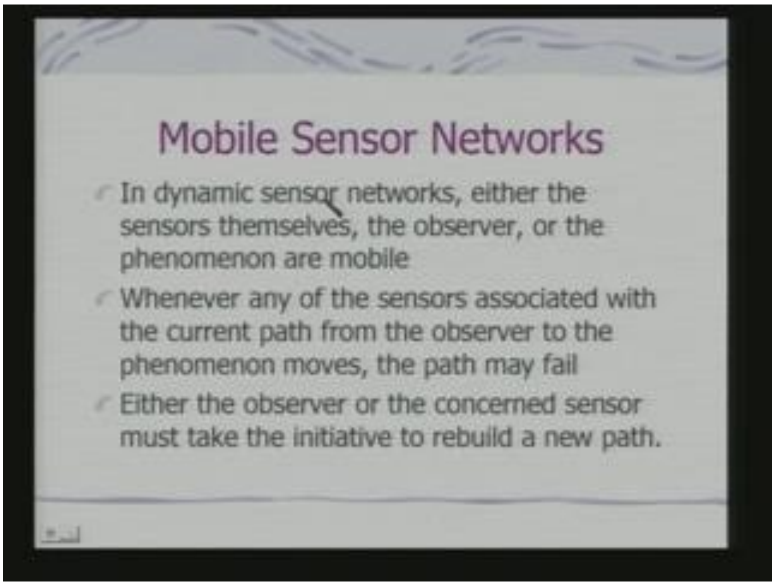
Static Sensor Networks

- In static sensor networks, there is no motion among communicating sensors, the observer and the phenomenon
- An selected node relays a summary of the local observations to the observer
  - Example: A group of sensors spread for temperature sensing
- Such algorithms can extend the lifetime of the sensor network because they trade-off local computation for communication.

So, the basis of this requirements are this objective we can distinguish sensor networks in to 2 classes static sensor networks and mobile sensor networks. In static sensor networks there is no motion among communicating sensors the observer and the phenomenon. So, everything is static and a selected node relays the summary of the local observations to the user. So, it can be a group of sensors spread for distributed temperature sensing. So, if I am interested in monitoring micro climating conditions. So, I can put temperature sensors distributed over a agricultural field and then I can have the data relate to the observer. Now, in this case they can be local processing at the node have also. So, if I say that a node is selected that realize at the summary of the local observation to the observer. Then what is happening the selected node is actually processing are computing the summary by accumulating the data from the local nodes.

So, I am not communicating therefore, the temperature data from each and every node to the observer. Because the communication caused typically would be more than that of computation cost if you talk about that designing an algorithm for extending of the life time of the sensor network, because the battery is the determining factor for this life time. Then I would like to have what is the aggregation a processing as done the local nodes themselves and then summary to be transmitted. So, the number of packets therefore, being transmitted to the observer reduces substantially and leads to energy conservation. So, this kind of algorithm can be used with reference to static sensor networks.

(Refer Slide Time: 17:00)



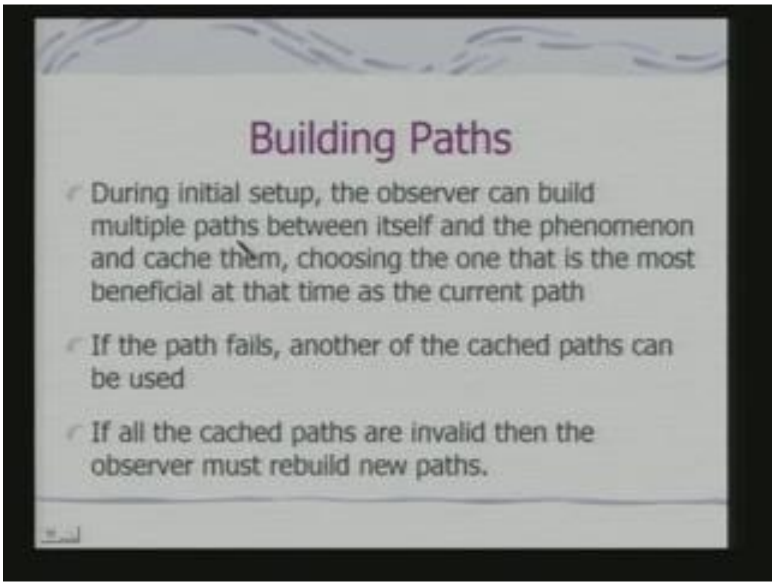
Mobile Sensor Networks

- In dynamic sensor networks, either the sensors themselves, the observer, or the phenomenon are mobile
- Whenever any of the sensors associated with the current path from the observer to the phenomenon moves, the path may fail
- Either the observer or the concerned sensor must take the initiative to rebuild a new path.

And the other hand we can have mobile sensor network or dynamic sensor networks. So, in dynamic sensor networks, either the sensors themselves, the observer, or the phenomena are mobile. So, there can be any one of this 3, because what I said as an network is characterized by sensor observer as well as the phenomena. So, any one of them in fact, they may be multiple of them can turn out to be mobile. So, whenever any of the sensor is associated with the current path from the observer to the phenomenon move the path may fail. So, what does the imply? That implies that path has to be recomputed.

So, I need to have a protocol which is adoptive to the topology of the network then either the observer or the concerned sensor must take therefore, initiative to rebuilt a new path. So, these becomes the basic adaptation of the overall system with respect to the changing topology. So, either the observer or the concern sensor must take the initiative to rebuild the path. So; that means, if the sensor has moved it should try to identify or that the sensors in its neighborhood if it is a broad casting message in the sensor in the whether there is other sensor in the neighborhood receiving the message or not has to be ascertained and the path has to be accordingly built.

(Refer Slide Time: 18:40)



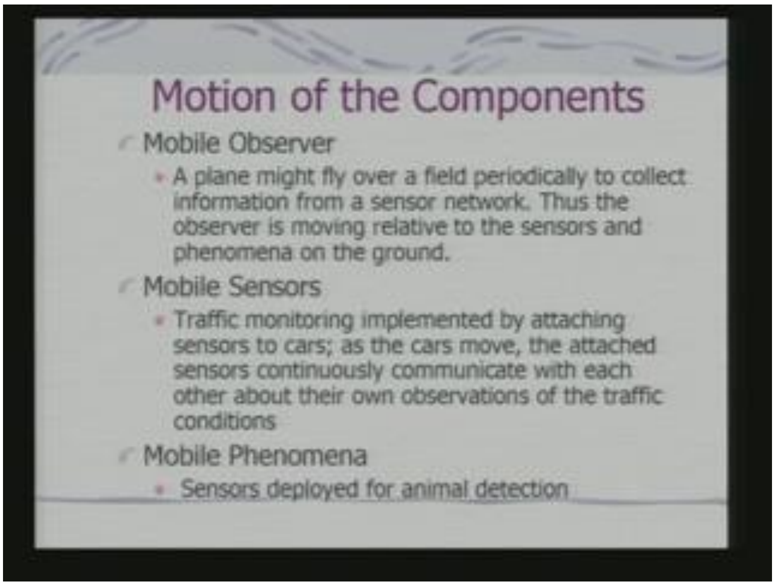
Building Paths

- During initial setup, the observer can build multiple paths between itself and the phenomenon and cache them, choosing the one that is the most beneficial at that time as the current path
- If the path fails, another of the cached paths can be used
- If all the cached paths are invalid then the observer must rebuild new paths.

.

So, building path is the basic scheme in this case during initial set up the observer can built multiple paths between itself and the phenomenon. And it can cache them that is we can make them enabled choosing the one that is the most beneficial at that time as the current path. So, what does this mean? It means that you can set up multiple paths multiple path for communication of information and at particular time. So, since you are set up the multiple paths the multiple path definitions are basically available with you. So, in the sense you have been cached. So, you are not creating the path on the fly the basic idea is created the path you have during the set up itself. And then what you are doing? You are choosing the one that is most beneficial at the particular time. So, if you find this path is not working out, you will switch to another path which has been already computed at the set up time itself. So, if the path fails another of the cache paths can be used if all the cache paths are invalid then the observer must built new paths. Obviously, we can realize the moment we built paths there is a delay involved. And that may have an adverse effect on your times synchronization then if you are actually observing a moving phenomenon.

(Refer Slide Time: 20:11)



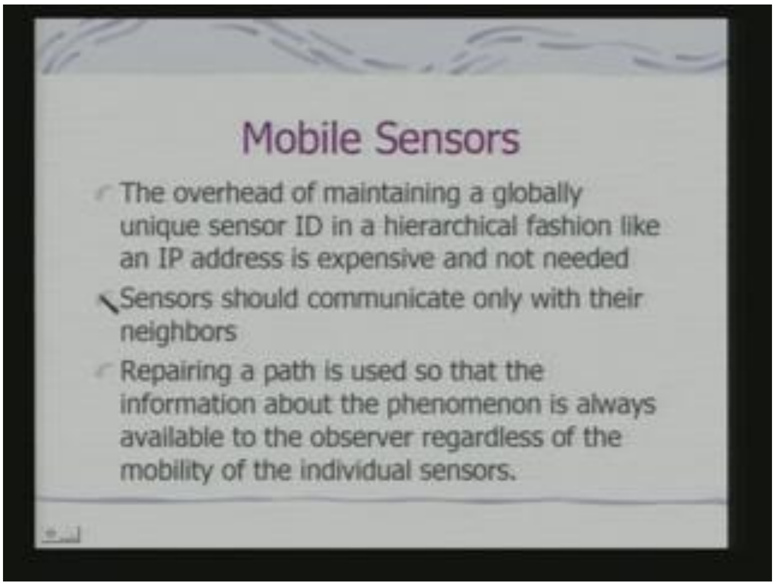
Motion of the Components

- Mobile Observer
  - A plane might fly over a field periodically to collect information from a sensor network. Thus the observer is moving relative to the sensors and phenomena on the ground.
- Mobile Sensors
  - Traffic monitoring implemented by attaching sensors to cars; as the cars move, the attached sensors continuously communicate with each other about their own observations of the traffic conditions
- Mobile Phenomena
  - Sensors deployed for animal detection

So, what are the kind of different motion? You have motion of the individual component. So, first is the mobile observer a plane might fly over periodically over a field periodically to collect information from a sensor network. The observer is moving relative to the sensors and phenomenon on the ground. Then you have got your mobile sensors let us say this is an example a traffic monitoring implemented per attaching sensors to cars. So, cars themselves are moving around cars can communicate with each other because they have got the sensors sensor nodes about their own observations of the traffic conditions.

So, on the basis of that you have to do the aggregation. So, you can realize at here the aggregation problem would be different from the case where you had a static sensor and the third gate moving around that is the phenomenon moving around. So, if you have a sensors deployed for animal detection the basic target tracking problem then the sensor is static and you had the target moving around. So, the aggregation problem it mean that of mobile phenomenon and that of mobile sensors could be therefore, different even that will be different when you have got a mobile observer. So, accordingly you are protocols for the network communication have to be adopted. So, when we have the mobile sensors.

(Refer Slide Time: 21:48)

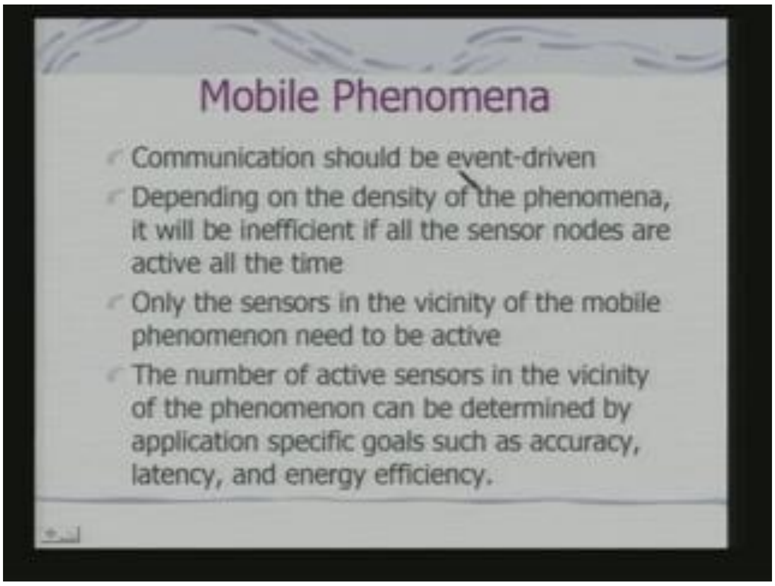


Mobile Sensors

- The overhead of maintaining a globally unique sensor ID in a hierarchical fashion like an IP address is expensive and not needed
- Sensors should communicate only with their neighbors
- Repairing a path is used so that the information about the phenomenon is always available to the observer regardless of the mobility of the individual sensors.

The overhead of maintaning a globally unique sensor ID in a hierarchical fashion like an IP address is expensive and not needed because your sensor is moving around. So, ip address is a hierarchical and uniquely associated with the node. So, this whole addressing scheme that we have familiar with that in the classical network, is no longer valid in this kind of a scenario. Sensor should communicate only with the neighbors the neighbors that are found and repairing a path is used. So, that the information about the phenomenon is always available to observer regardless of the mobility of the individual sensor. So, the sensor is moving around and its finds a neighbor. So, through a set of neighbors a path has to be rebuilt for getting the data back to the observer fine. So, what we say is that the movement are mobile nodes moves around the currently existing path no longer remain valid. So, the path needs to be repaired since the path needs to be repaired. So, what does it mean? That means, you identify in a set of node through which the message is to be relate when it is a mobile phenomenon typically the communication should be event driven.

(Refer Slide Time: 23:14)



Mobile Phenomena

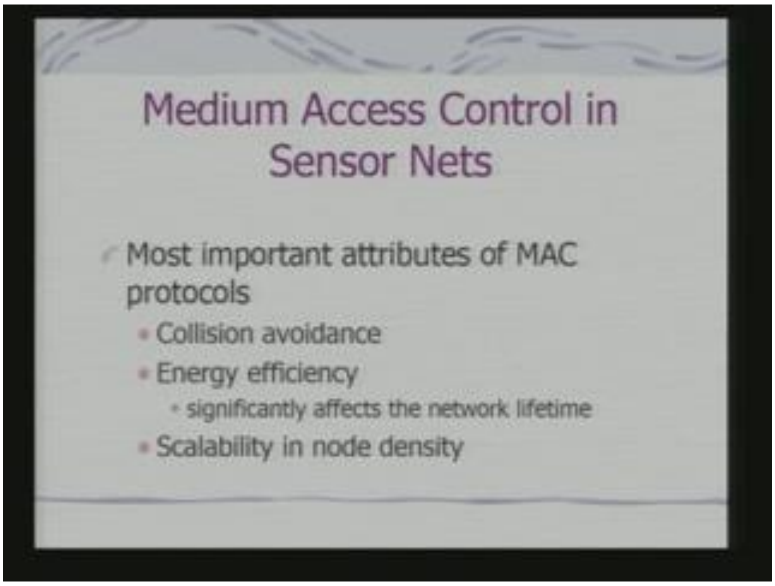
- Communication should be event-driven
- Depending on the density of the phenomena, it will be inefficient if all the sensor nodes are active all the time
- Only the sensors in the vicinity of the mobile phenomenon need to be active
- The number of active sensors in the vicinity of the phenomenon can be determined by application specific goals such as accuracy, latency, and energy efficiency.

That means you are tracking a ((refer time: 23:19)) tracking a target. So, whenever a node detects the target that is an event. So, the whole communication therefore, should be set up in an event driven fashion and depending on the density of the phenomena it will be inefficient if all the sensor nodes are active all the time. Because you have just consider in a forest you are put in this sensors all around it is not necessary for all the sensors to be actively monitoring. It is necessary only for those sensors which have just got the triggering event and that triggering event may trigger other sensors in a synchronized fashion. So, only the sensors in the vicinity of the mobile phenomenon needs to be active. So, therefore, you can realize the design of such sensors also becomes an issue because typically an external event has to trigger the sensors.

So, normally what will happen such nodes will be slipping when the target is in the vicinity that is the sensor senses? The target that will generate here the sensor I am referring to this sensing element sensing element getting the target. Then what will do? It will generate an interrupt to activate the system which can now start processing the data is it is clear otherwise there is an unnecessary wastage of resource. The number of sensors in the vicinity of the phenomenon can be retrieving by application specific goals such a active raze latten share energy efficiency. Because on the basis of which you have to design the deployment pattern, because the deployment pattern is something very very important why because it is a phenomenon, what you are trying to do?

But putting the sensors the doing the space time sampling of the phenomenon and if you are doing an under sampling of the space time phenomenon you are not making correct observations is it is clear. So, that matching of your sensor deployment strategy that is the sampling intervals space time sampling interval should be such that the matches that of the phenomenon otherwise you will meats out the basic features of the phenomenon itself. So, now with this basic background we shall not going to details of this variety of this protocols, but we shall look at some of the issues and some of the basics of some of the protocols and the designs which are relevant to sensor network. But; obviously, we do not have contain to get in to their details.

(Refer Slide Time: 26:05)

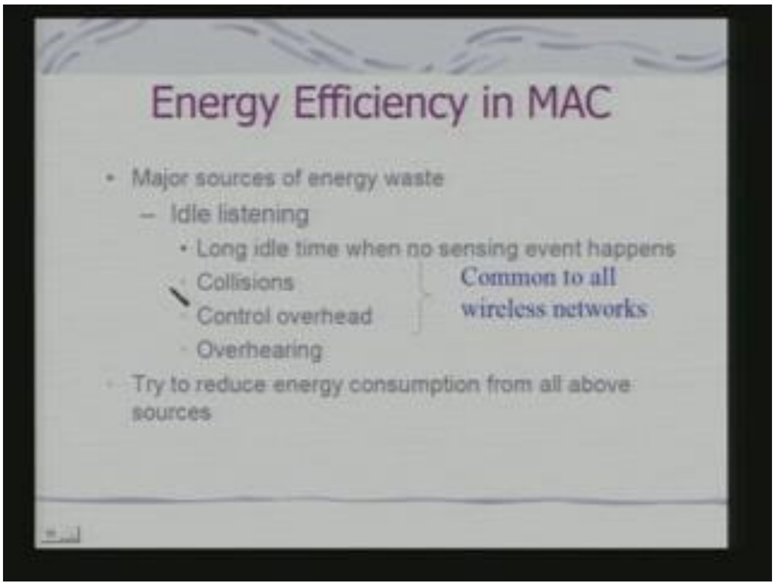


Medium Access Control in Sensor Nets

- Most important attributes of MAC protocols
  - Collision avoidance
  - Energy efficiency
    - significantly affects the network lifetime
  - Scalability in node density

So, what is the key issues in medium access control in sensor networks? The key should be collision avoidance and energy efficiency as well as scalability in terms of node density. So, energy efficiencies significantly affects the network lifetime. So, if you look in to it; obviously, what we are talking about? I would like to avoid the collision the 2 nodes in the vicinity should not start broadcasting at the same point in time. We should like to have energy efficiency, because that significantly affects the network lifetime and the system should be scalable. That means, if I you increase the node density the network traffic should not be search that actually the channel get chocked the communication channel gets chocked effectively I am not being able to transfer the data. So, such a scenario should not happen. So, let us look at 1 aspect of it than that of energy efficiency.

(Refer Slide Time: 27:16)

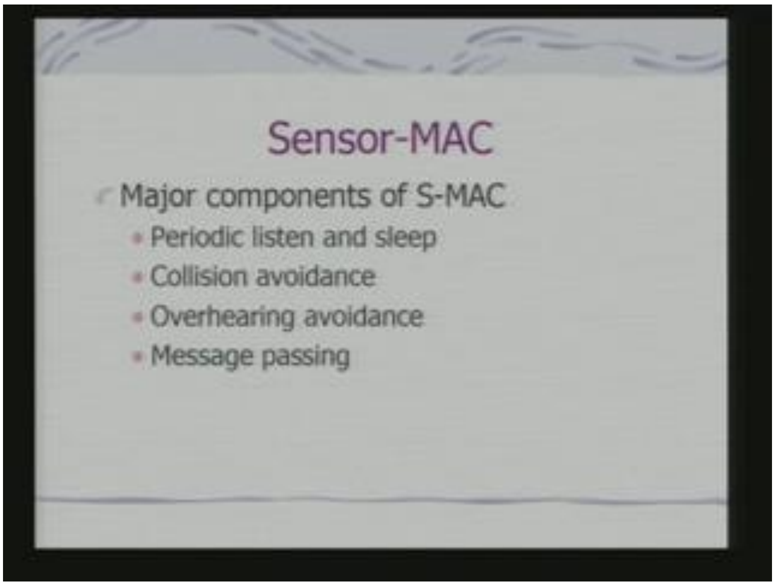


Energy Efficiency in MAC

- Major sources of energy waste
  - Idle listening
    - Long idle time when no sensing event happens
    - Collisions
    - Control overhead
    - Overhearing
- Try to reduce energy consumption from all above sources

Now, what we have found the common to all wireless network the major sources of energy wastage is idle listening, because you have got long idle time when no sensing event happens this is typical to that of the sensor network. But you actually do an idle listening to find out whether there is a collision or not. If you are using a typical collision avoidance protocol what you do you actually sends that whether there is any communication are talking place or not. If it is not taking place then you wait for your interval and then you start communicating that is exactly what is done in 822.11. So, in this case the listening part is actually the most energy consuming part, because you are listening to the channel. So, the long idle time when there is no sensing event happens if the node continuous to sense the channel it will be wasting energy. So, what we need? We need to have a strategy to reduce energy consumption from all above sources. So, therefore, is a idle time to listening and collision about avoidance control overhead and also over hearing; that means, the message which is not intended for you should not here as well. So, how to design such a protocol? So, we shall look at very preemptive protocol and basic aspect of its protocol.

(Refer Slide Time: 28:53)

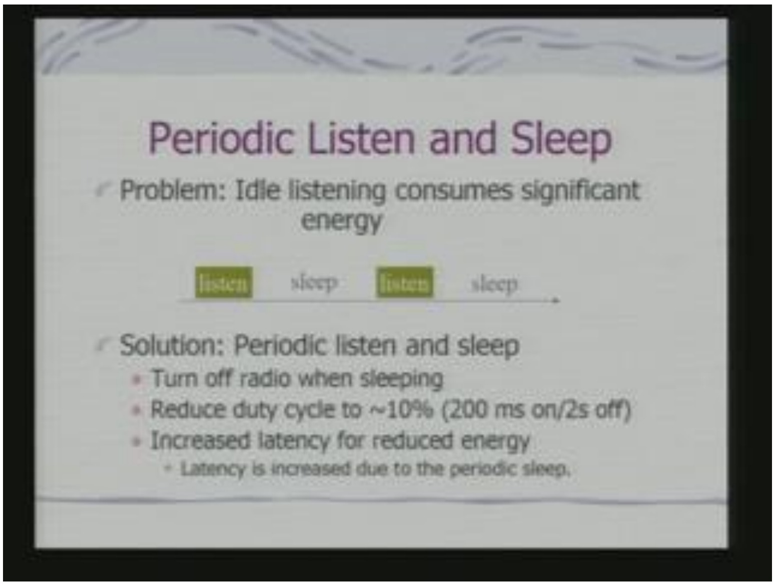


Sensor-MAC

- Major components of S-MAC
  - Periodic listen and sleep
  - Collision avoidance
  - Overhearing avoidance
  - Message passing

So, this is called sensor MAC or SMAC. So, the key issue here is periodic listen and sleep and using the periodic listen and sleep you can do collision avoidance overhearing avoidance as well as message passing. In fact, the major part that is are these the periodic listen and sleep collision avoidance overhearing avoidance and message passing. We shall look at this phenomenon of periodic listen and sleep which distinguishes this MAC protocol from that of your 822.11 protocol which is for wireless line.

(Refer Slide Time: 29:37)



Periodic Listen and Sleep

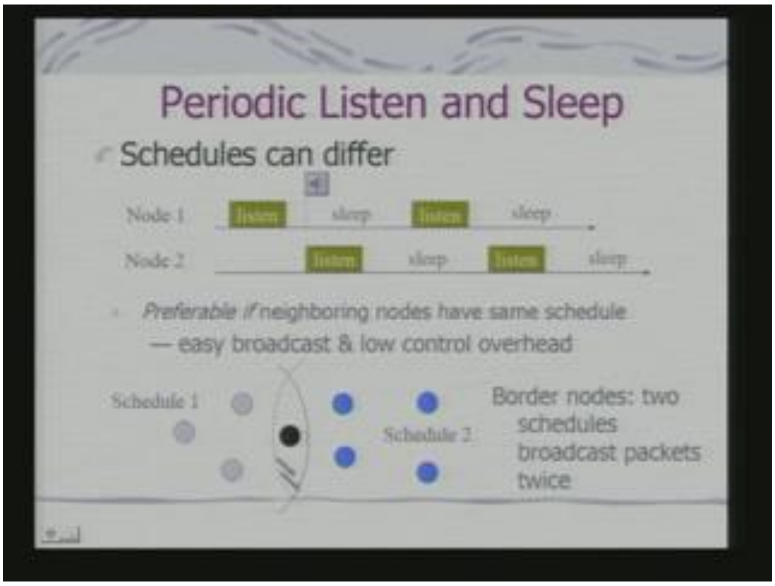
- Problem: Idle listening consumes significant energy

listen
sleep
listen
sleep

- Solution: Periodic listen and sleep
  - Turn off radio when sleeping
  - Reduce duty cycle to ~10% (200 ms on/2s off)
  - Increased latency for reduced energy
    - Latency is increased due to the periodic sleep.

So, here if you see that what we are looking at the, we are trying to have a phase or a period when the node listens a period when the node sleep then it listens then it sleeps. So, this node that is always listening it is listening and sleep now what is the difference if you see with the Bluetooth; Bluetooth was they where definite slots available at which your slaves you are communicating with the master. So, there are the basic concept of master and slave, but here we are not talking about a concept of master and slave in a way it is a absolutely symmetric configuration with each node configure as such that there is no distinguishing between master and slave. So, each nodes follows a periodic schedule it has a fix period for listening and fix period for sleeping this is the basic SMAC protocol. So, turn off radio when sleeping. So, saves energy and it reduces duty cycle to about of the order of 10 percent so, about 20 millisecond on and 2 seconds off. So, it is a big saving although it implies there is a increase latency for reduced energy latency is increased due to periodic sleep. So, therefore, how the how the whole communication takes place. So, you are listening.

(Refer Slide Time: 31:12)



Periodic Listen and Sleep

- Schedules can differ

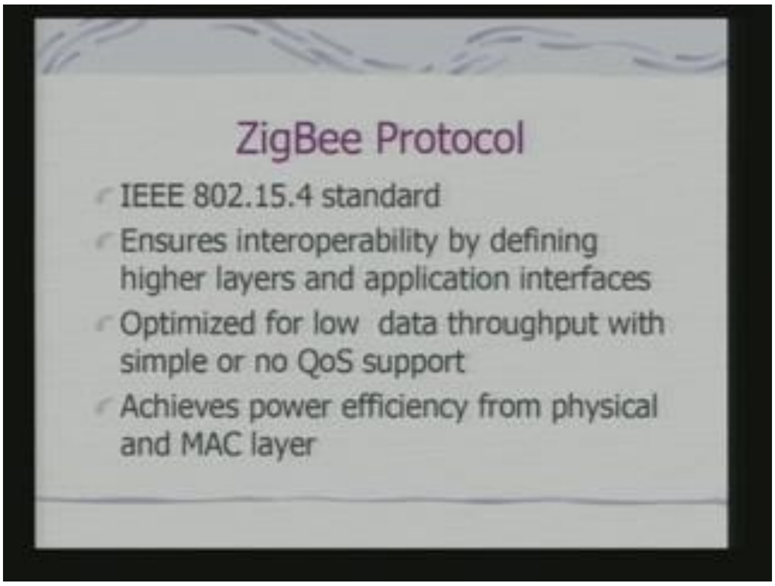
- *Preferable if* neighboring nodes have same schedule
  — easy broadcast & low control overhead

Border nodes: two schedules broadcast packets twice

So, after listening you can to a broadcast if I having basically the neighboring nodes these neighboring nodes can have different schedules the nodes can have different schedules. So, when you are broadcasting at this point the node this node 2 is in a listen node. So, it can listen and receive the data and since they are in different modes; obviously, this neighboring node will not broadcast at that point because it is listening. So, naturally your collision gets avoided. Now, So, if you say that preferably is neighboring nodes have same schedule than also the same issue comes up. So, you have got easy broad cast and low control overheads. So, if you have the same schedule that is both of them are listening at the same point. So, you know their listening and accordingly you can act. Now, they can be different schedules followed by a network for a different set of nodes. So, if you look in to this I am considering this node as a border node. So, border nodes can follow 2 schedules to broadcast packets twice because when if you look in to it there was 1 point of time here I am broadcasting packets.

So, this network when these packets are broadcasted these nodes are following a sleep schedules. So, since they are sleeping the packet cannot be received. So, I have to have a another broadcast. So, the neighboring nodes following different schedule actually increases in a way control overhead, because you have to do a multiple broadcast. So, what we say it is a preferable if may varying nodes have similar schedule, but if I know exactly. So, I the other issue is what I illustrated here that if have these kind of periods defined then this gets also naturally lick to avoidance of collisions no 2 nodes will be really trying to broadcast at the same point in time is its clear. So, this is the basic philosophy of the SMAC or sensor MAC protocol now; obviously, you can realize since it is a periodic listen and slip at each node there is a substantial saving in the energy. And the nodes are not really listing to the channel to avoid detect collision and that is the major source of energy consumption.

(Refer Slide Time: 33:52)

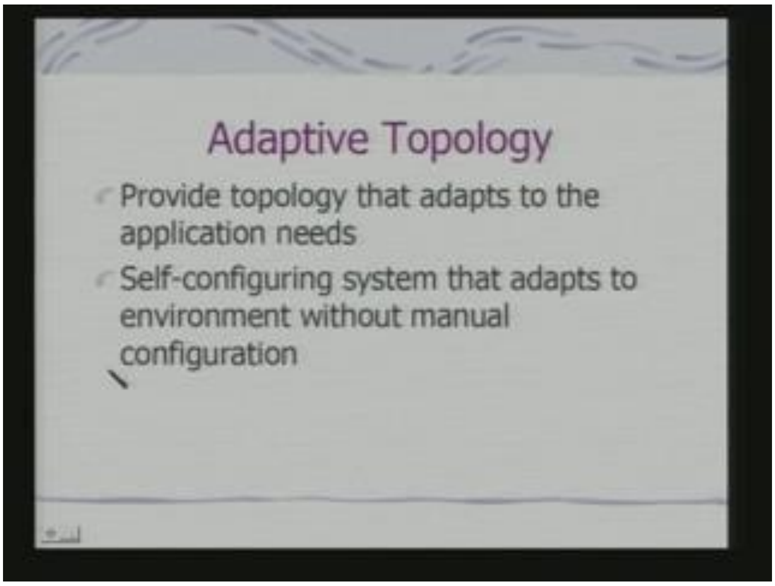


ZigBee Protocol

- IEEE 802.15.4 standard
- Ensures interoperability by defining higher layers and application interfaces
- Optimized for low data throughput with simple or no QoS support
- Achieves power efficiency from physical and MAC layer

There is a zigbee protocol which is which is actually a different from this which is currently immerging IEEE standard for this kind of sensor networks. And zigbee refers to an industry ((refer time; 34:08)) which ensures interoperability by defining higher layers and application interfaces. And it is optimized for low data throughput with simple or no Qos supports. And achieves power efficiency from physical and MAC layer in a sensor MAC I have talked about the power efficiency primarily from MAC layer point of view. But from the physical layer point of view also you specify your communication pattern that your usage of physical layer that can also lead to energy saving.

(Refer Slide Time: 34:40)

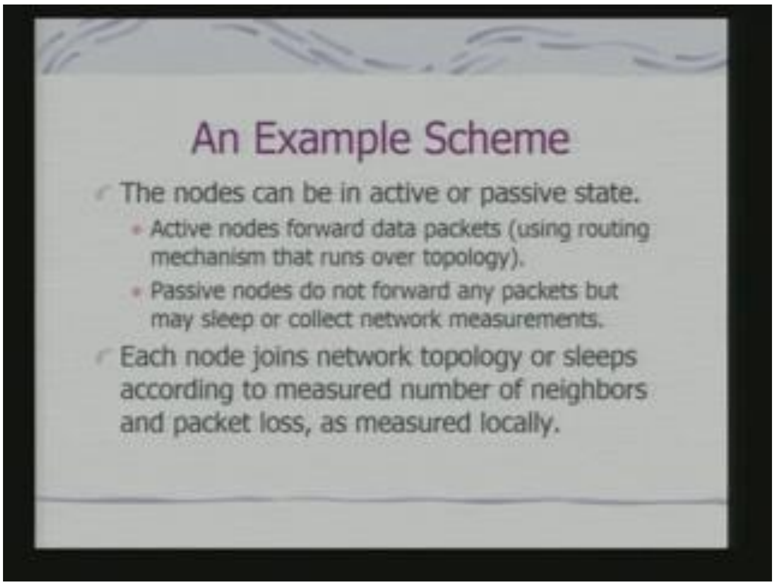


Adaptive Topology

- Provide topology that adapts to the application needs
- Self-configuring system that adapts to environment without manual configuration

Next lets you can an adaptive topology. So, adoptive topology basically provides a mechanism to adopt the existing topology with respect to the application needs. I have already illustrated that concept that when the nodes are moving around and any to establish a path. So, this kind of adaptation is essentially. So, what we say is that we have got a self configuring system that adapts to environment without manual configuration. So, these self configurations can refer to even in such cases where there is a failure of a node being detected even when a node fails whether the adapt node can take up how the other node can take up? They can take up provided the actually move around if the nodes have got the around mobility else they can also take up if they change may be there clipping pattern. So, let us look at a simple mechanism to deal with this.

(Refer Slide Time: 35:44)



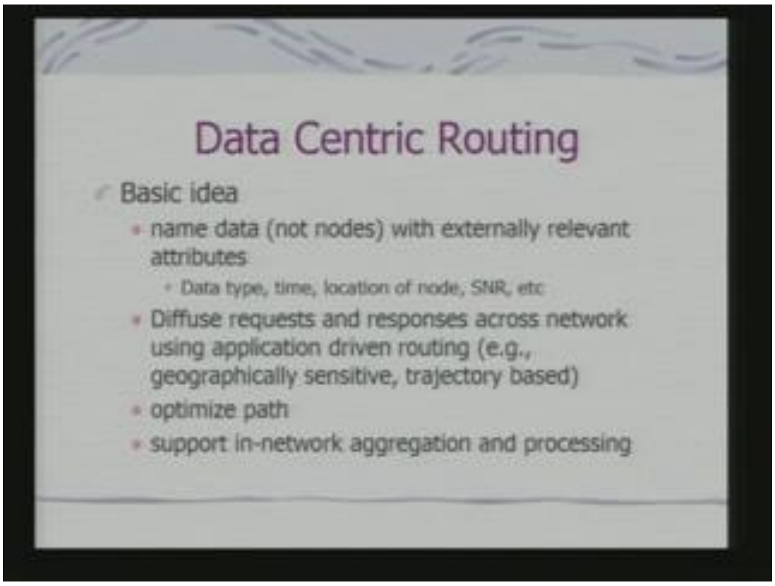
An Example Scheme

- The nodes can be in active or passive state.
  - Active nodes forward data packets (using routing mechanism that runs over topology).
  - Passive nodes do not forward any packets but may sleep or collect network measurements.
- Each node joins network topology or sleeps according to measured number of neighbors and packet loss, as measured locally.

So, we say that the nodes can be in active or passive state nodes which are in active state over data packet which using a routing mechanism that runs over the topology. Passive nodes do not forward any packets, but may sleep or collect network measurements fine. So, now each nodes joins network topology or sleeps according to measured number of neighbors and packet loss as measured locally. So, here there is an adaptation of the topology with no node movements only making nodes active and inactive. So, I got a set of nodes and nodes are either in active or in passive state and; obviously, in this case what we what we can impose? We can impose greatest sleeping period of the nodes in a MAC it was fix periods of listen and sleep. If I know a node is currently not participating in forwarding a packet then the node can continue to sleep even if it has now the time to listen because it is not participating at all.

So, each node checks whether you have got a certain neighbors in its neighborhood that is sensors neighbors as well as it sensors when there is a packet loss if there is a packet loss; that means, there may be a maul function of a neighbor. So, to take care of it maul function it should activate itself. So, there is a therefore, an automotive decision and autonomous decision at the node level itself to decide whether to become active or passive. So, it is not just determine by the MAC protocol, but determine from the point of your application that is of packet transfer the other interesting feature of this sensor network say the high level is data centric routing.

(Refer Slide Time: 37:49)



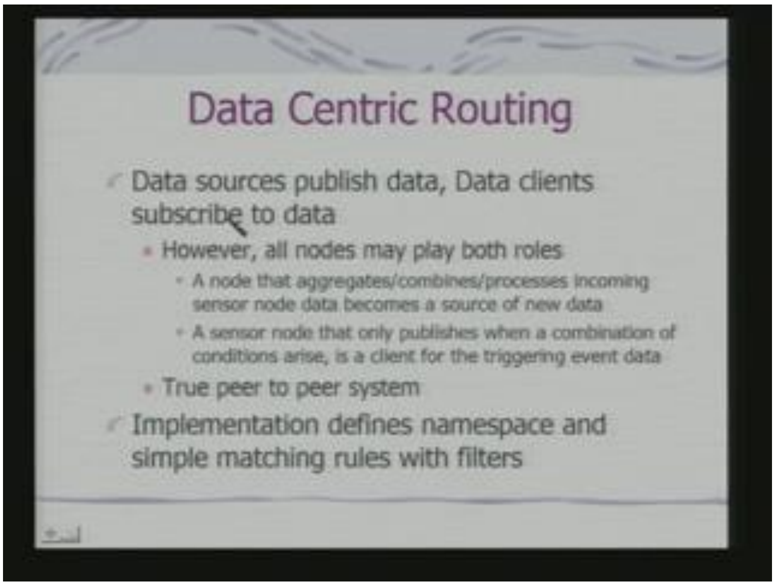
Data Centric Routing

- Basic idea
  - name data (not nodes) with externally relevant attributes
    - Data type, time, location of node, SNR, etc
  - Diffuse requests and responses across network using application driven routing (e.g., geographically sensitive, trajectory based)
  - optimize path
  - support in-network aggregation and processing

So, typically you have seen routing is what routing means I rout packets from source to destination and there can be also intermediate nodes. Now, in the data centric routing we say that we name data and not nodes with externally relevant attribute and not nodes with externally relevant attributes. We name data with externally relevant attribute to if the data is being transfer across the network. That means, data with diffusing to the network any node which needs the data can make use of this attributes and start using the data. So, these as a phenomenon is different from that of the pure address centric routing where the data is targeted for a node having a specific address that address need not be an IP address some form of address. So, what happens here is you diffuse request and responses across network using application driven routing. So, this application driven routing can be geographically sensitive can be trajectory based because I might like to rout the packets along the trajectory.

The example of tracking is a typically a case can I have detected a particular target and I have got the information on the, to the packet and trying to send it along you optimize the path. And these would support in network aggregation and processing because if you have the data packet and moving along a track. And along the track if actually the target is moving then at each node you can do what you can aggregate the newly sense data wit already sense data. So, that aggregation takes place in the network itself and here when the data is being send are being broadcasted it is not really being send to an address, but being broadcasted with a tag describing the data. So, the other node which are listening to it they can really process the data depending on the request that is being defused. So, what we say that the data sources publish data and data clients subscribe to data.

(Refer Slide Time: 40:06)

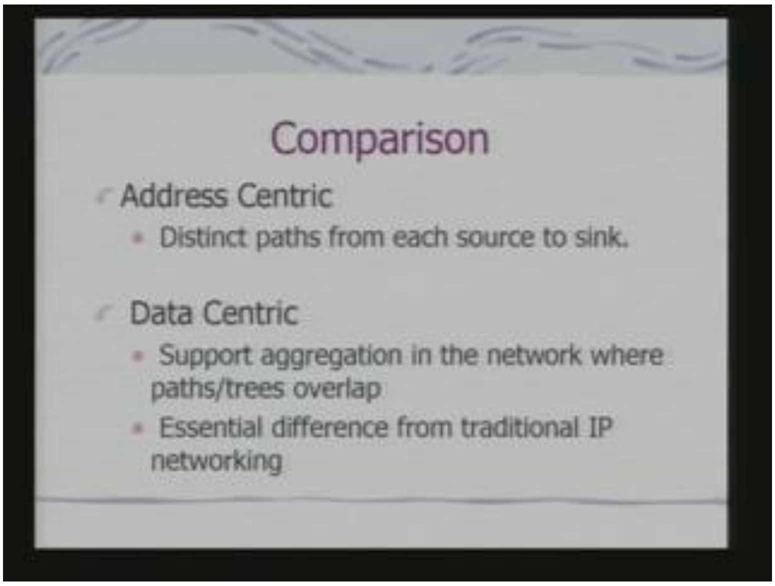


Data Centric Routing

- Data sources publish data, Data clients subscribe to data
  - However, all nodes may play both roles
    - A node that aggregates/combines/processes incoming sensor node data becomes a source of new data
    - A sensor node that only publishes when a combination of conditions arise, is a client for the triggering event data
  - True peer to peer system
- Implementation defines namespace and simple matching rules with filters

And all nodes may play both the roles fine. So, a node that aggregates combines processes incoming sensor data become a source of new data. A sensor node that only publishes when a combinations of conditions arise is a client for triggering event data and it becomes a true peer to peer system the implementation define namespace. And simple matching rules with filters what we need matching rules with filters, because I need to match the attribute of the data to know whether it is to be processed. So, it is not targeted for a particular node, but node understands whether it requires to process the data depending on the application request that it is processing. So, the basic node is that of publication of the data that is publishing of the data as well as consumption of the data de3pending on the demand.

(Refer Slide Time: 41:06)

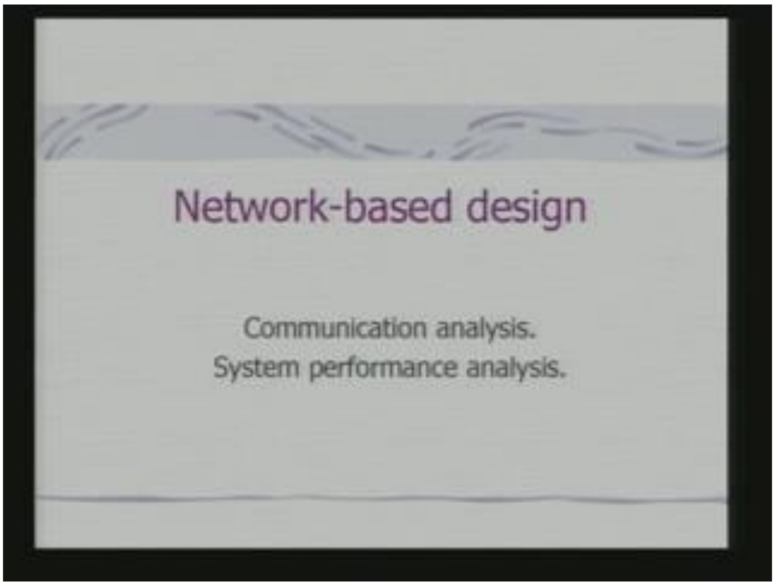


Comparison

- Address Centric
  - Distinct paths from each source to sink.
- Data Centric
  - Support aggregation in the network where paths/trees overlap
  - Essential difference from traditional IP networking

So, what we have in terms of comparison, because the data centric routing, routing is you will not find in a classical network classical network is always address centric routing. So, in an address centric routing each path is from source to sink. So, you have distinct path from source to sink. In a data centric routing support aggregation in the network where paths or tress overlap there may be multiple paths. So, that data centric processing can take place over such multiple paths and this is the essential difference from traditional IP network. So, this data centric routing which is absolutely\ important for observing the phenomenon in a sensor network makes the whole routing exercise completely different from IP centric networks. So, this more are less finishes our basic characteristics of sensor networks now you shall look at the issue of designing such a network base system.

(Refer Slide Time: 42:12)

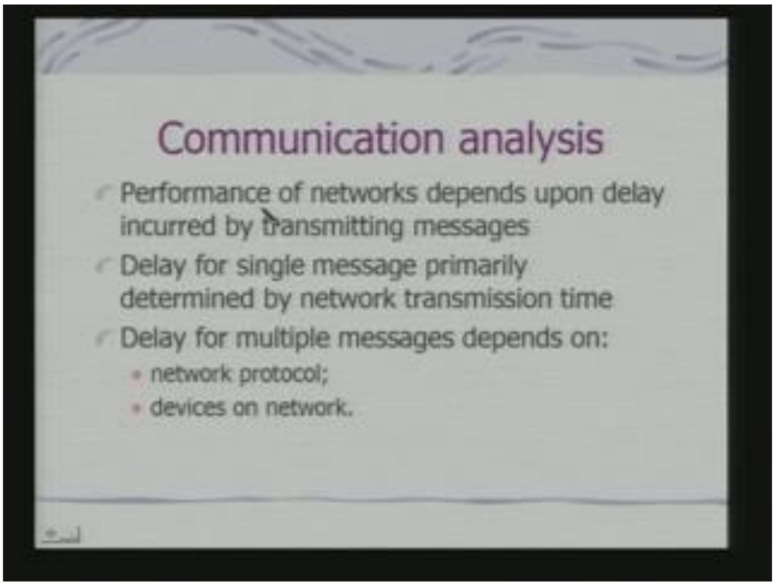


Network-based design

Communication analysis.
System performance analysis.

Designing of such a network base system independent of what protocol being used involves 2 aspects. One is communication analysis and system performance analysis in fact, these aspects are important for designing either sensor networks may be a network of processors to go in to SOC may be a network of embedded system to put on a automobile. So, any such distributed, embedded system design needs to do communication analysis as well as system performance analysis. So, let us try to understand them in a quality defeasance not really in a analytical way.

(Refer Slide Time: 43:05)

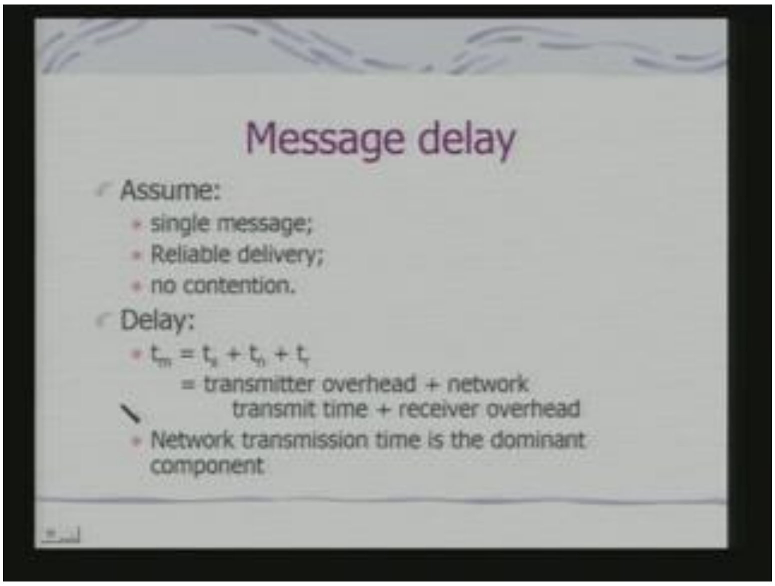


Communication analysis

- Performance of networks depends upon delay incurred by transmitting messages
- Delay for single message primarily determined by network transmission time
- Delay for multiple messages depends on:
  - network protocol;
  - devices on network.

So, in a communication analysis; obviously, the performance of networks depends up on delay incurred by transmitting messages and delay for single message primarily determine by the network transmission time. But delay for multiple messages depends on networks protocols devices on the network, because when you are looking at a single message you can assume or reliable delivery of a single message and you are not considering the multiple messages being delivered. So, multiple messages whether they will be schedule for delivery or not also depends on the protocol being followed. It will also depend on the different paths the message is following. So, then the delay for multiple message is becomes difficult to analyze also and delay for single message is primarily determine by the network transmission time.

(Refer Slide Time: 44:05)

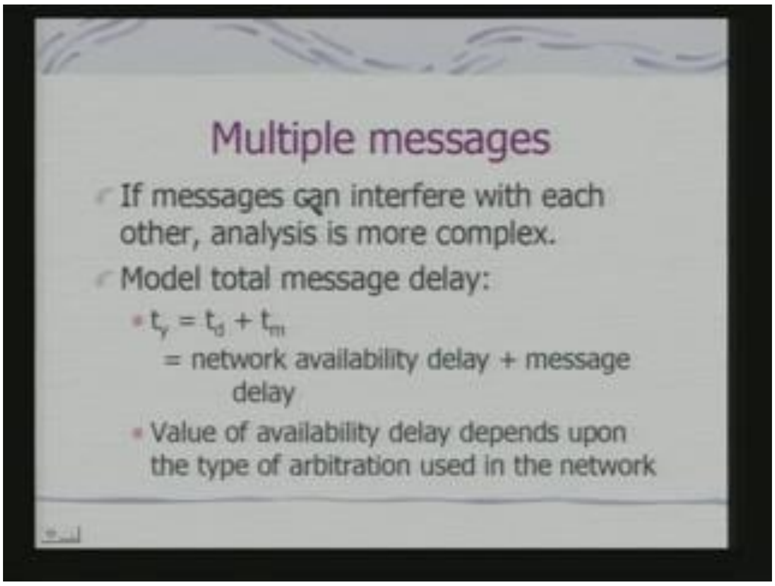


Message delay

- Assume:
  - single message;
  - Reliable delivery;
  - no contention.
- Delay:
  - $t_m = t_x + t_n + t_r$
    = transmitter overhead + network transmit time + receiver overhead
  - Network transmission time is the dominant component

Now, if you look at this that a single message a single message reliable delivery and no contention. Then the delay can be calculated by this formula this is the transmitter time this is the network transmit time and receiver overhead and a typically the network transmission time dominates the other 2 factors. So, for any design if you have the estimate of these factor that would enable you to take decisions.

(Refer Slide Time: 44:39)

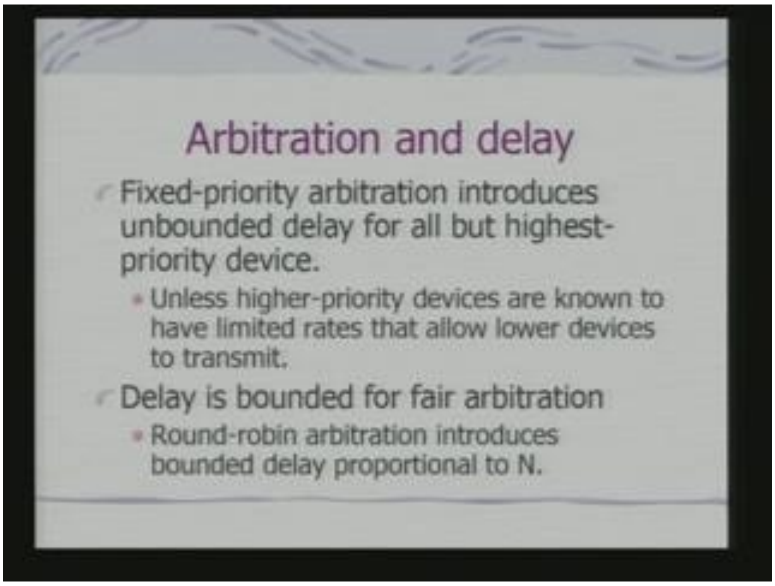


Multiple messages

- If messages can interfere with each other, analysis is more complex.
- Model total message delay:
  - $t_y = t_d + t_m$
    = network availability delay + message delay
  - Value of availability delay depends upon the type of arbitration used in the network

So, let us take a simple example of a multiple we shall look at a example if multiple processors when there involved in to a in a network for a distributed computing task. On the other hand if you consider multiple messages. The messages can interfere with each other and analysis becomes; obviously, more complex. So, we need to module the message delay. So, there are 2 aspects of its; 1 is our network availability of delay and the message delay the message delay is what we had already seen. Network availability delay becomes critical when we are looking at multiple messages, because this is the availability of the network what is availability of the network means that the communication channel is being made available to the message for being transmitted. Now, So, value of availability delay depends upon the type of arbitration used in the network because arbitration will decide whether the communication channel will be available for transmitting a message or not.

(Refer Slide Time: 45:43)

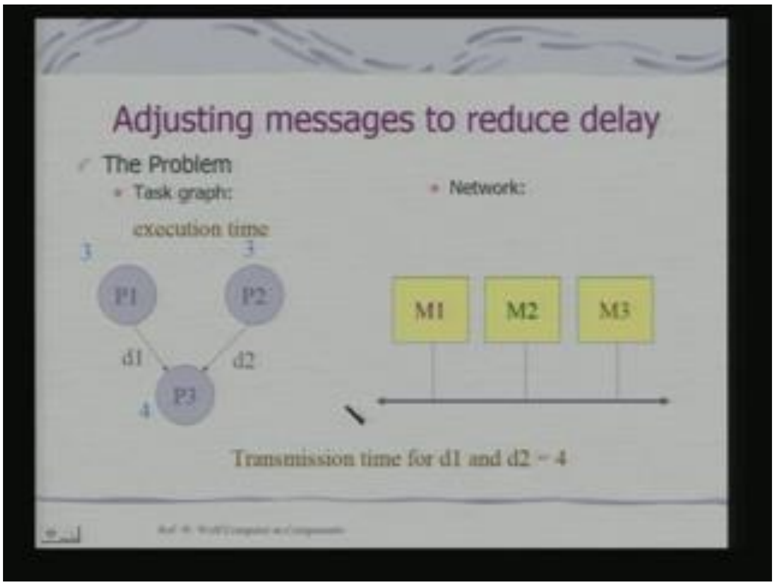


Arbitration and delay

- Fixed-priority arbitration introduces unbounded delay for all but highest-priority device.
  - Unless higher-priority devices are known to have limited rates that allow lower devices to transmit.
- Delay is bounded for fair arbitration
  - Round-robin arbitration introduces bounded delay proportional to N.

So, there can be various forms of arbitration you have already seen when we studied this buses and as well as network. There are various forms of arbitration which are used if you use a fix priority arbitration then you get what is called an unbounded delay. If you have an unbounded delay you cannot really design a real time system on such a network, because you cannot guarantee the task being finished in a deadline. So, unless a higher priority devices are known to have a limited rates that allow lower devices to transmit. You may have absolutely unbounded delay for the low priority devices and delay, but; obviously, you can see if I have the fix priority. If I have a tasks with deadlines then if they are given the high priority then you can definitely satisfy those tasks. But with tasks with the low priority can suffer from unbounded delay and the deadlines may not be met. Now, delay is bounded for fear arbitration. So, very fear common arbitration policies round robin a round robin arbitration introduces bounded delay proportional to N why? Why should be proportional to N if there are N packets containing for the bandwidth? Because if is given in term each packet or each source is provided with a slot in turn. So; obviously, the delay would be proportional to N that is the number of packets containing for the slot.

(Refer Slide Time: 47:17)



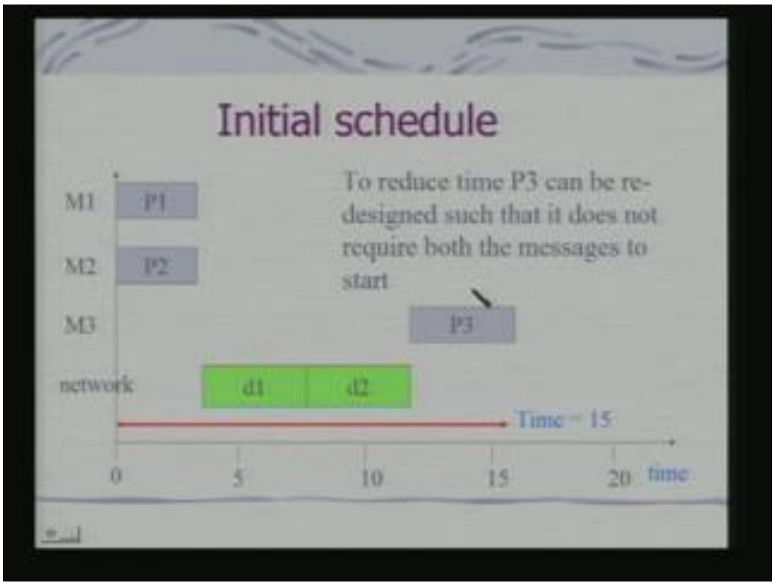
Adjusting messages to reduce delay

- The Problem
  - Task graph:
  - Network:

Transmission time for d1 and d2 = 4

So, let us took an example let us have a look at a example and with respect to an example understand that what to do with this delay whether we can really redesign some aspects of the network to increase efficiency once we know this delays. The problem is indicated by this task graph. So, this are the tasks P 1 P 2 and the produce data d 1 and d 2 which is required by P 3 to complete the tasks and you are using a network are processors to finish this job. So, there are 3 processors M 1 M 2 and M 3 . So, first you need a task mapping we have done this task mapping lets we have done this tasks mapping. Once we have this task mapping this task will take different times to be executed on this processing elements and then we need to communicate this. So, we I have shown the basic execution time say 3 units it does not a matter what is the time units? 3 units of time for P 1 and P 2, and 4 units for the transmission time. Now, if this is the set up then what will be the total time required for completion of these tasks?

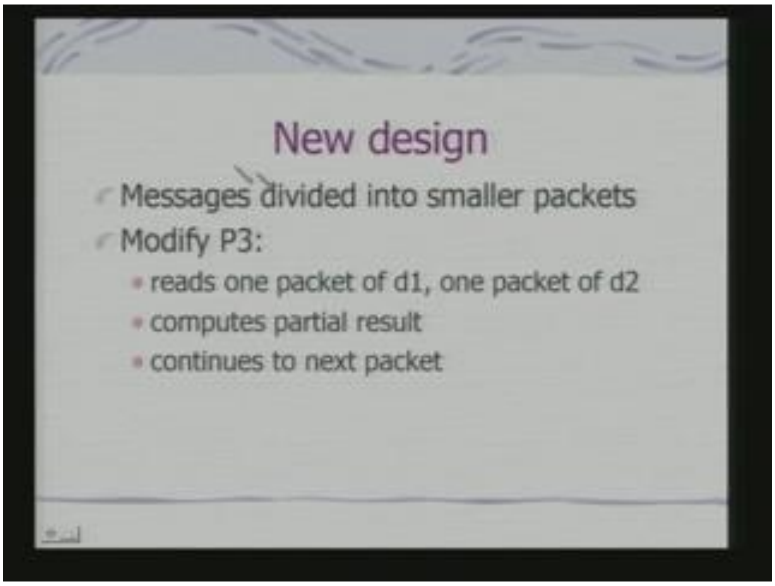
(Refer Slide Time: 48:34)


Initial schedule

To reduce time P3 can be re-designed such that it does not require both the messages to start

So, I can draw this I can have P 1 and P 2 scheduled concurrently, because there on different processing elements, but I cannot schedule P 3 simply, because P 3 can only start provided receives the data d 1 and d 2. And these data will take this amount of time the total execution time in this case would be 15. Now, the question is can I redesign the whole network base system to reduce the time? So, the question is to reduce the time P 3 can redesign such that it does not require the messages to start or may be some part of the messages to start. So, it becomes the question redesigning task P 3 this is basically your design problem.

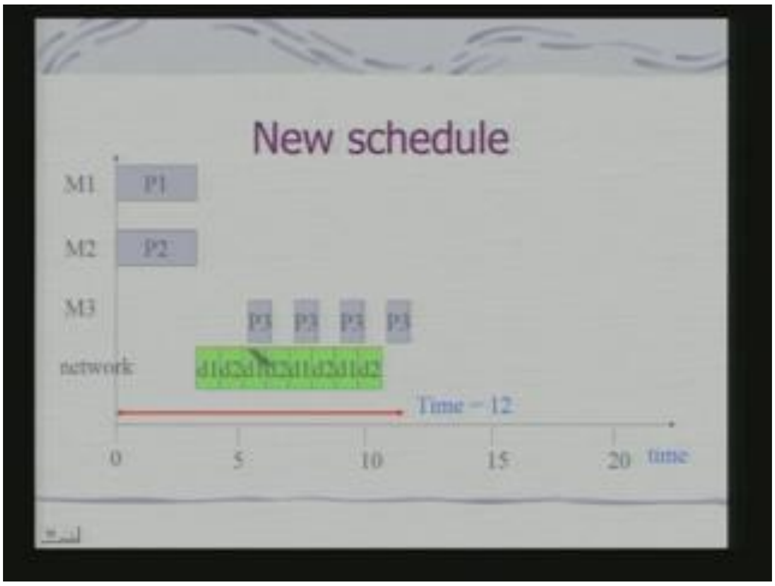
(Refer Slide Time: 49:21)


New design

- Messages divided into smaller packets
- Modify P3:
  - reads one packet of d1, one packet of d2
  - computes partial result
  - continues to next packet

So, what we do this same is messages can be divided into smaller packets. So, the definition of the packet is a critical component here and you modify P threes such that you reads 1 packet of d 1 and 1 packet of d 2 and computes partial results and continuous to next packet. So, therefore, what it means that you can now schedule P 3 overlap with actual transmission, because earlier P 3 was getting scheduled only after transmission was complete.

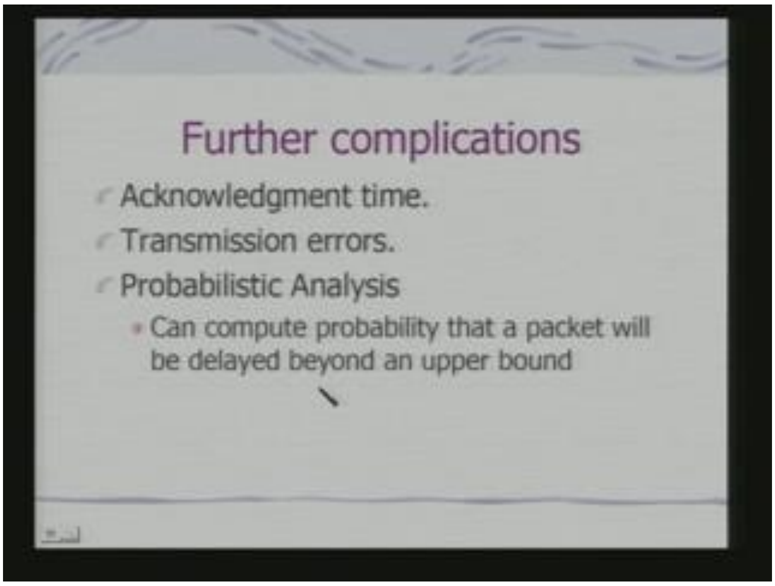
(Refer Slide Time: 49:51)



New schedule

So, in this case I have divided the data in to packets. So, these are d 1 d 2 correspond to packets of from task P 1 or P 2. So, once d 1 and d 2 is arrived P 3 can be scheduled. So, P 3 now can be overlap the next packets from 1 and 2. So, this can continue and. So, I can say that the P 3 can finish in time sat 12 units because it has been now overlap with that of your packet transmission. So, this becomes a issue of design these is an important aspects when you are looking at designing a network or processor or in SOC when even when you are designing a set of sensor nodes to monitor a phenomena. Because if you see all your collaborative data processing and signal processing tasks would have this kind of task dependence as well. And what is the data packet which have be broadcasted or transmitted that has to be decided to have tasks performing in such a way that they meet the time constraint of the phenomenon.

(Refer Slide Time: 51:09)

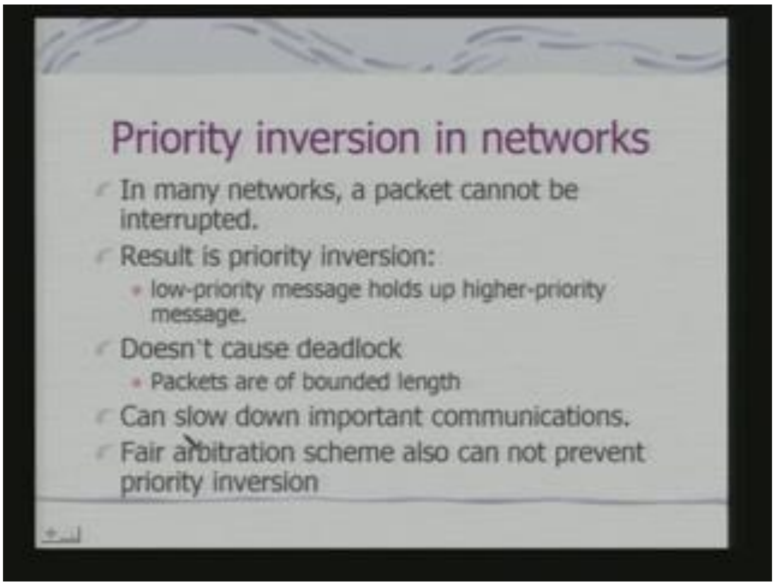


Further complications

- Acknowledgment time.
- Transmission errors.
- Probabilistic Analysis
  - Can compute probability that a packet will be delayed beyond an upper bound

They can be further complication, because you have not taken in to account acknowledgement time and transmission errors. So, that can be taken care of to do a probabilistic analysis or a simulation. So, you can compute a probability that a packet will be delayed beyond an upper bound. So, that kind of a analysis makes this design process more sophisticated.

(Refer Slide Time: 51:28)



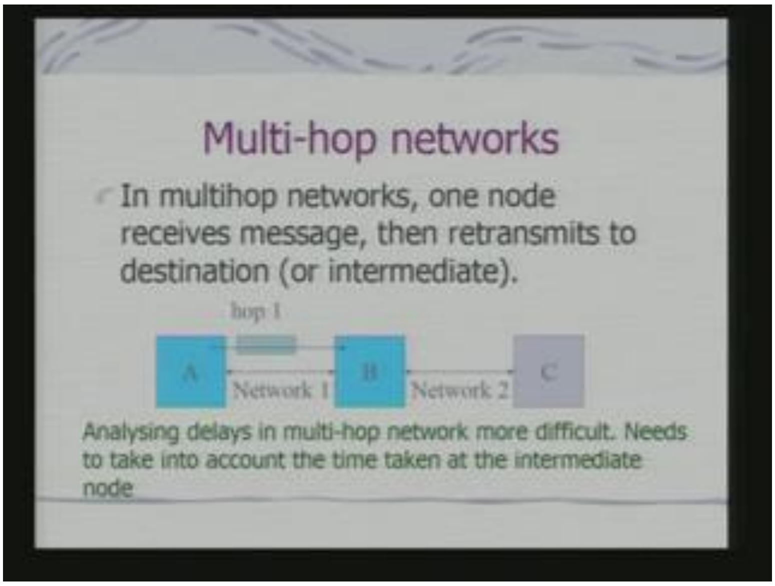
Priority inversion in networks

- In many networks, a packet cannot be interrupted.
- Result is priority inversion:
  - low-priority message holds up higher-priority message.
- Doesn't cause deadlock
  - Packets are of bounded length
- Can slow down important communications.
- Fair arbitration scheme also can not prevent priority inversion

There is also important phenomenon which happens in network base system is that of priority inversions in many networks. The packet cannot be interrupted in almost all the networks a packet cannot be interrupted a processes can be interrupted, but a packet cannot be interrupted. So, what happens? A low priority message can hold up a high priority message even if you have a high priority that cannot be send since the low priority packet is currently transmission, but this on cause any kind of the deadline because packets are bounded links. So, packets will definitely get delivered. So, the next 1 will get a chance, but what happens is a priority inversion.

We have seen priority inversion in the context of process scheduling here we are looking at priority inversion in the context of communication packet scheduling. So, priority inversion would lead to what will slow down important communication. So, you need to know that how to deal with this slewing down in fact, fair arbitration scheme also cannot prevent priority inversion, because packets cannot be packet transmission cannot be interrupted. So, that may lead to design of separate networks a network for a higher priority tasks a network for should not have a higher priority tasks. Higher priority messages or network may be for low priority messages.

(Refer Slide Time: 53:01)



Multi-hop networks

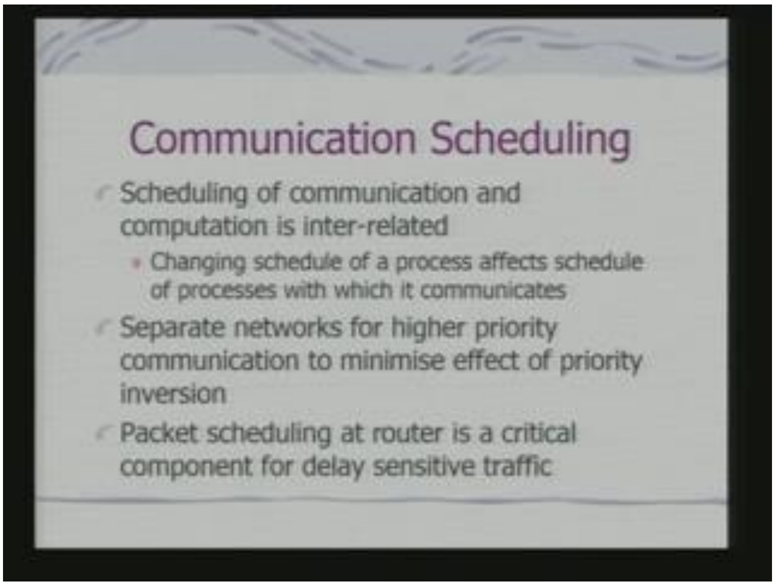
- In multihop networks, one node receives message, then retransmits to destination (or intermediate).

hop 1
A
Network 1
B
Network 2
C

Analysing delays in multi-hop network more difficult. Needs to take into account the time taken at the intermediate node

In case of a multi hop networks one node receives messages then retransmit to destination or intermediate. So, in analyzing delay in multi hop network; obviously, much more difficult and next to take in to account that time taken at intermediate nodes, because the intermediate node depending on the tasks could decide what to schedule and when to schedule.

(Refer Slide Time: 53:21)

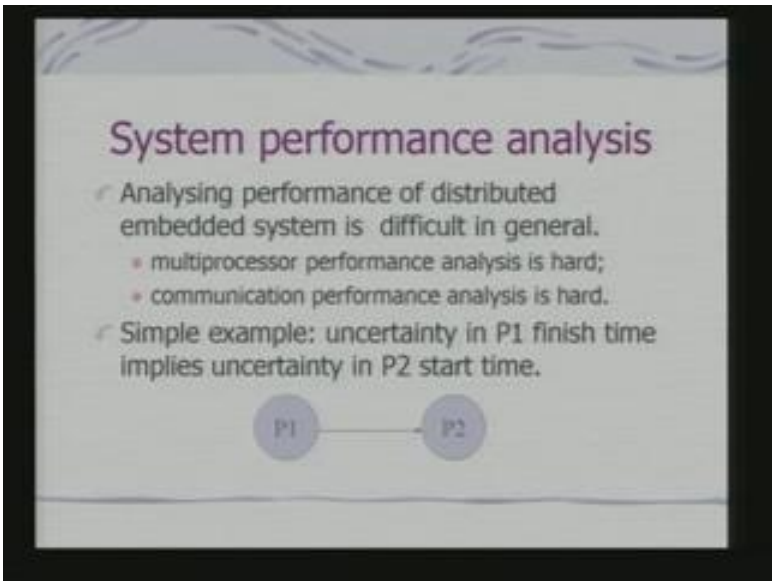


Communication Scheduling

- Scheduling of communication and computation is inter-related
  - Changing schedule of a process affects schedule of processes with which it communicates
- Separate networks for higher priority communication to minimise effect of priority inversion
- Packet scheduling at router is a critical component for delay sensitive traffic

So, what we therefore, coming to the problem of communication scheduling, but scheduling of communication and computation is inter related. Changing schedule of a process affects schedule of processors with which it communicates, because that process can not start until analyze it gets a data from a processes. And you may need to have to take care of the delay, because of priority inversion separate networks for higher priority communication to minimize the effect of priority inversion. And a packet scheduling at router therefore, becomes a critical components for delay sensitive traffic. Because the delay the packet would suffer depends on how the router decides to schedule which scheduling of a packet means what which packet being selected what transmission that is allocation of communication channel. Now, related to this is the system performance analysis.

(Refer Slide Time: 54:21)

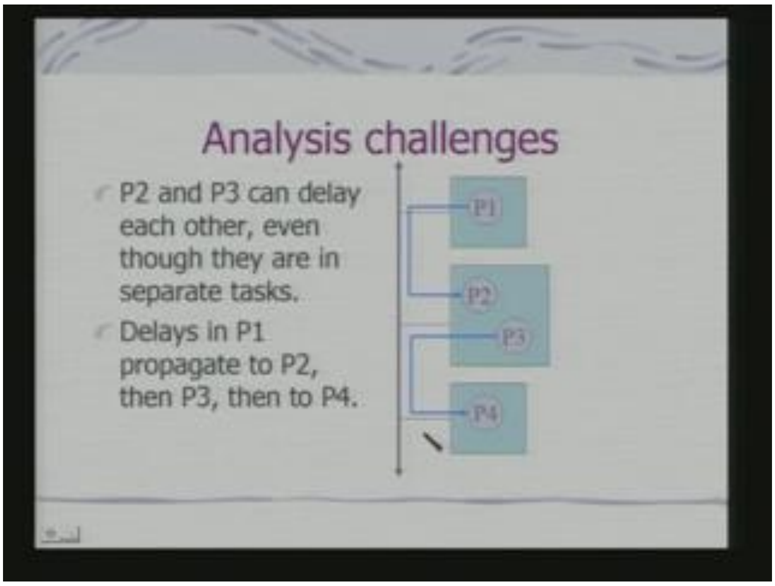


System performance analysis

- Analysing performance of distributed embedded system is difficult in general.
  - multiprocessor performance analysis is hard;
  - communication performance analysis is hard.
- Simple example: uncertainty in P1 finish time implies uncertainty in P2 start time.

Analyzing performance of distributed embedded system is difficult in general, because multiprocessor involve and communication performance is also informed. Simple example, here is uncertainty in P 1 finish time implies uncertainty in P 2 start time take an example again

(Refer Slide Time: 54:40)

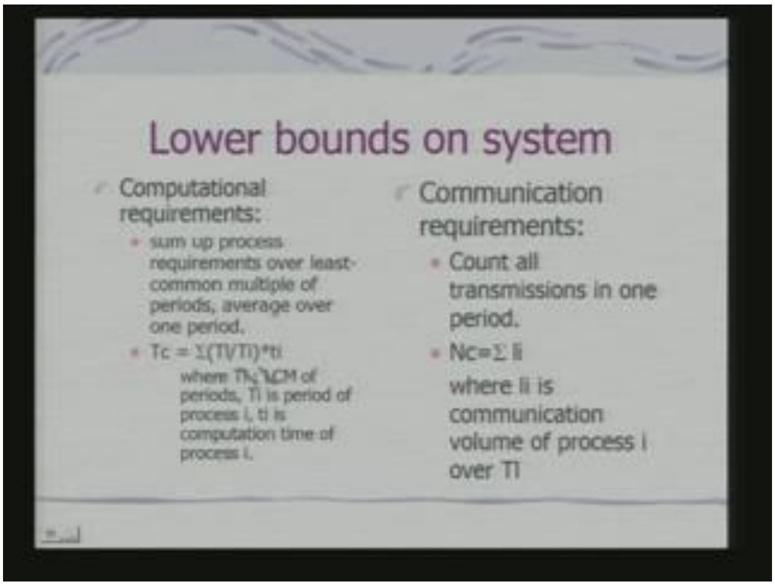


Analysis challenges

- P2 and P3 can delay each other, even though they are in separate tasks.
- Delays in P1 propagate to P2, then P3, then to P4.

Here I have got 3 processor and I have got independent task P 1 and P 2 belongs to 1 tasks set P 3 and P 4 belong to another task set. Now, P 2 and P 3 mapped on to 1 processor. So, depending on scheduling of P 2 and P 3 P 4 and P 1 as well as performance of P 1 would critically determine P 2. Because if P 1 is delayed P 2 will be delayed if P 2 is delayed P 3 may not be schedule delayed P 3 may not be schedule P four may be delayed. So, P 2 and P 3 can delay each other even though they are in separate tasks . So, delays in P 1 propagate to P 2 then P 3 then to P 4. So, these kind of analysis becomes importantly to be done. So, scheduling policy of the processors themselves decides the scheduling policy of the tasks as well. So, if you look at the communication requirements.

(Refer Slide Time: 55:38)

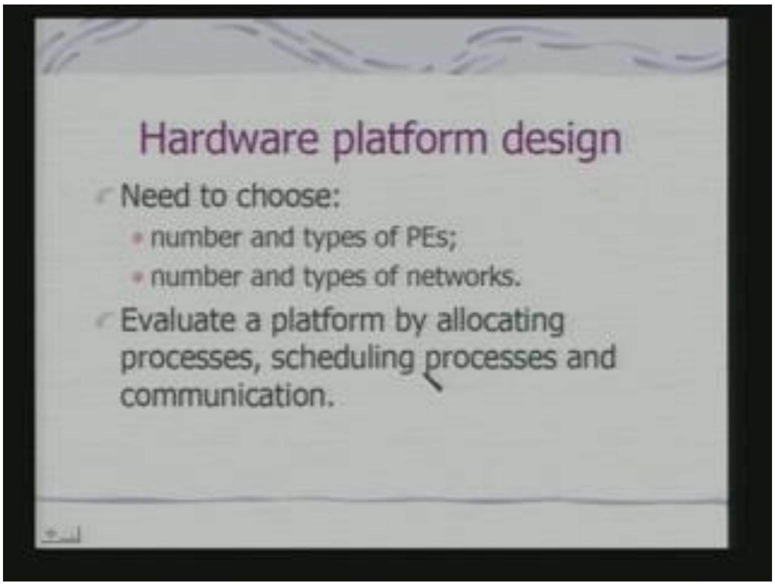


Lower bounds on system

- Computational requirements:
  - sum up process requirements over least-common multiple of periods, average over one period.
  - Tc = Σ(Tl/Ti)*ti
    where Tl=LCM of periods, Ti is period of process i, ti is computation time of process i.
- Communication requirements:
  - Count all transmissions in one period.
  - Nc=Σ li
    where li is communication volume of process i over Tl

So, this is the kind of bound you will get. So, these bound comes from say if TL is the LCM of the periods of the tasks because many of the tasks is the periodic tasks LCM will give me the least common multiple of this periods. So, if TI is the period of each tasks then if I divide this I get the number of times a task get schedule I know this as a processing time of a individual tasks. If I submit of that gives me the lower bound of the computational requirement and a lower bound of the communication requirement comes from summation of the LI, because Li is the communication volume of process I over TL, because TL is the LCM of the period. So, that gives me the lower bound and the communication time. So, all analysis and design has to be based on this parameters.

(Refer Slide Time: 56:33)

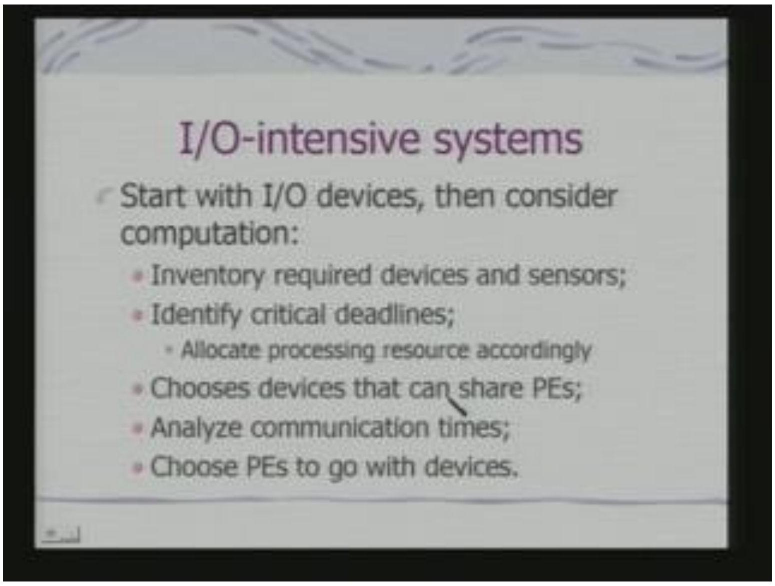


Hardware platform design

- Need to choose:
  - number and types of PEs;
  - number and types of networks.
- Evaluate a platform by allocating processes, scheduling processes and communication.

So, when I am designing therefore, on the basis of these parameters we need to choose processing elements and the types of networks. And we evaluate a platform by allocating processes scheduling processes and communication such that this requirements are satisfied.

(Refer Slide Time: 56:50)

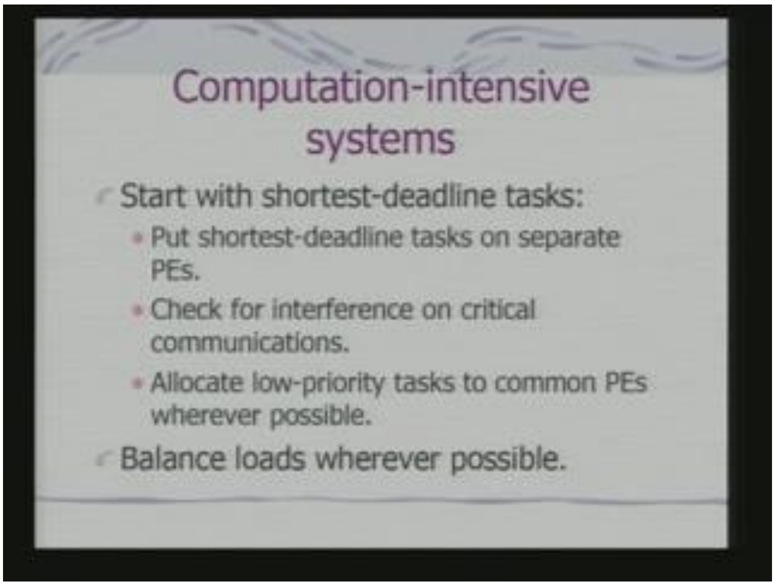


I/O-intensive systems

- Start with I/O devices, then consider computation:
  - Inventory required devices and sensors;
  - Identify critical deadlines;
    - Allocate processing resource accordingly
  - Chooses devices that can share PEs;
  - Analyze communication times;
  - Choose PEs to go with devices.

So, if you look at the IO intensive systems which is typically may be your sensor network. So, you need to have the devices and sensors identify the critical deadlines allocating a processing resource accordingly and analyze your communication times and then choose PEs to go with devices. So, different PEs can go with different devices depending on your deadline if you doing a computational intensive tasks.

(Refer Slide Time: 57:20)

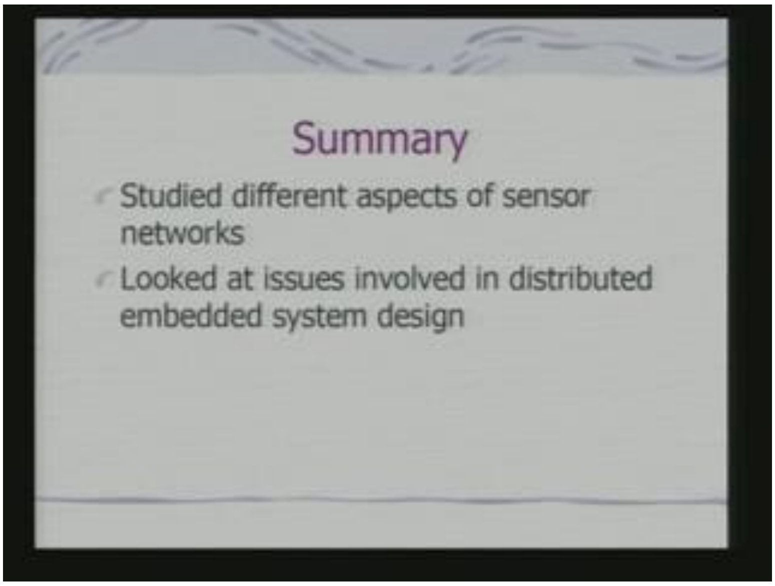


Computation-intensive systems

- Start with shortest-deadline tasks:
  - Put shortest-deadline tasks on separate PEs.
  - Check for interference on critical communications.
  - Allocate low-priority tasks to common PEs wherever possible.
- Balance loads wherever possible.

You should start with the shortest deadline tasks and that should be put on different PEs. So, I am talking about the distributed system. So, they should be put on different PEs and check for interference and critical communications allocate low priority tasks to common PEs wherever possible and you need to balance loads. So, this is a typically computation intensive system. So, this brings as to the end of our discussions on network embedded system.

(Refer Slide Time: 57:54)



Summary

- Studied different aspects of sensor networks
- Looked at issues involved in distributed embedded system design

So, we have studied today different aspects of sensor networks and looked at issues involved in designing of distributed embedded system. In the next class onwards we shall basically look at this design issues in more detail.